



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'dracut.conf.5'

\$ man dracut.conf.5

DRACUT.CONF(5) dracut DRACUT.CONF(5)

NAME

dracut.conf - configuration file(s) for dracut

SYNOPSIS

/etc/dracut.conf /etc/dracut.conf.d/*.conf

/usr/lib/dracut/dracut.conf.d/*.conf

DESCRIPTION

dracut.conf is loaded during the initialisation phase of dracut.

Command line parameter will override any values set here.

*.conf files are read from /usr/lib/dracut/dracut.conf.d and

/etc/dracut.conf.d. Files with the same name in /etc/dracut.conf.d will

replace files in /usr/lib/dracut/dracut.conf.d. The files are then read

in alphanumerical order and will override parameters set in

/etc/dracut.conf. Each line specifies an attribute and a value. A #

indicates the beginning of a comment; following characters, up to the end of the line are not interpreted.

dracut command line options will override any values set here.

Configuration files must have the extension .conf; other extensions are

ignored.

`add_dracutmodules+=` <dracut modules> "

Add a space-separated list of dracut modules to call when building the initramfs. Modules are located in `/usr/lib/dracut/modules.d`.

`force_add_dracutmodules+=` <dracut modules> "

Force to add a space-separated list of dracut modules to the default set of modules, when host-only mode is specified. This parameter can be specified multiple times.

`omit_dracutmodules+=` <dracut modules> "

Omit a space-separated list of dracut modules to call when building the initramfs. Modules are located in `/usr/lib/dracut/modules.d`.

`dracutmodules+=` <dracut modules> "

Specify a space-separated list of dracut modules to call when building the initramfs. Modules are located in `/usr/lib/dracut/modules.d`. This option forces dracut to only include the specified dracut modules. In most cases the "add_dracutmodules" option is what you want to use.

`add_drivers+=` <kernel modules> "

Specify a space-separated list of kernel modules to add to the initramfs. The kernel modules have to be specified without the ".ko" suffix.

`force_drivers+=` <list of kernel modules> "

See `add_drivers` above. But in this case it is ensured that the drivers are tried to be loaded early via `modprobe`.

`omit_drivers+=` <kernel modules> "

Specify a space-separated list of kernel modules not to add to the initramfs. The kernel modules have to be specified without the ".ko" suffix.

`drivers+=` <kernel modules> "

Specify a space-separated list of kernel modules to exclusively include in the initramfs. The kernel modules have to be specified without the ".ko" suffix.

`filesystems+=` <filesystem names> "

Specify a space-separated list of kernel filesystem modules to exclusively include in the generic initramfs.

```
drivers_dir="<kernel modules directory>"
```

Specify the directory where to look for kernel modules.

```
fw_dir+=" :<dir>[:<dir> ...] "
```

Specify additional colon-separated list of directories where to look for firmware files.

```
libdirs+=" <dir>[ <dir> ...] "
```

Specify a space-separated list of directories where to look for libraries.

```
install_items+=" <file>[ <file> ...] "
```

Specify additional files to include in the initramfs, separated by spaces.

```
install_optional_items+=" <file>[ <file> ...] "
```

Specify additional files to include in the initramfs, separated by spaces, if they exist.

```
compress="{cat|bzip2|lzma|xz|gzip|lzo|lz4|zstd|<compressor [args ...]>}"
```

Compress the generated initramfs using the passed compression program. If you pass it just the name of a compression program, it will call that program with known-working arguments. If you pass arguments, it will be called with exactly those arguments.

Depending on what you pass, this may result in an initramfs that the kernel cannot decompress. To disable compression, use "cat".

```
squash_compress="{<compressor [args ...]>}"
```

Compress the squashfs image using the passed compressor and compressor specific options for mksquashfs. You can refer to mksquashfs manual for supported compressors and compressor specific options. If squash module is not called when building the initramfs, this option will not take effect.

```
do_strip="{yes|no}"
```

Strip binaries in the initramfs (default=yes).

```
aggressive_strip="{yes|no}"
```

Strip more than just debug symbol and sections, for a smaller initramfs build. The "do_strip=yes" option must also be specified (default=no).

do_hardlink="{yes|no}"

Hardlink files in the initramfs (default=yes).

prefix=" <directory> "

Prefix initramfs files with <directory>.

hostonly="{yes|no}"

Host-only mode: Install only what is needed for booting the local host instead of a generic host and generate host-specific configuration (default=no).

hostonly_mode="{sloppy|strict}"

Specify the host-only mode to use (default=sloppy). In "sloppy" host-only mode, extra drivers and modules will be installed, so minor hardware change won't make the image unbootable (e.g. changed keyboard), and the image is still portable among similar hosts.

With "strict" mode enabled, anything not necessary for booting the local host in its current state will not be included, and modules may do some extra job to save more space. Minor change of hardware or environment could make the image unbootable.

hostonly_cmdline="{yes|no}"

If set to "yes", store the kernel command line arguments needed in the initramfs. If hostonly="yes" and this option is not configured, it's automatically set to "yes".

hostonly_nics+=" [<nic> [<nic> ...]] "

Only enable listed NICs in the initramfs. The list can be empty, so other modules can install only the necessary network drivers.

persistent_policy="<policy>"

Use <policy> to address disks and partitions. <policy> can be any directory name found in /dev/disk. E.g. "by-uuid", "by-label"

tmpdir="<temporary directory>"

Specify temporary directory to use.

Warning

If chrooted to another root other than the real root device, use

--fstab and provide a valid /etc/fstab.

```
use_fstab="{yes|no}"
```

Use /etc/fstab instead of /proc/self/mountinfo (default=no).

```
add_fstab+=" <filename> "
```

Add entries of <filename> to the initramfs /etc/fstab.

```
add_device+=" <device> "
```

Bring up <device> in initramfs, <device> should be the device name.

This can be useful in host-only mode for resume support when your swap is on LVM an encrypted partition.

```
mdadmconf="{yes|no}"
```

Include local /etc/mdadm.conf (default=no).

```
lvmconf="{yes|no}"
```

Include local /etc/lvm/lvm.conf (default=no).

```
fscks=" <fsck tools> "
```

Add a space-separated list of fsck tools. If nothing is specified, the default is: "umount mount /sbin/fsck* xfs_db xfs_check xfs_repair e2fsck jfs_fsck reiserfsck btrfsck". The installation is opportunistic (non-existing tools are ignored).

```
nofscks="{yes|no}"
```

If specified, inhibit installation of any fsck tools (default=no).

```
ro_mnt="{yes|no}"
```

Mount / and /usr read-only by default (default=no).

```
kernel_cmdline="parameters"
```

Specify default kernel command line parameters.

```
kernel_only="{yes|no}"
```

Only install kernel drivers and firmware files (default=no).

```
no_kernel="{yes|no}"
```

Do not install kernel drivers and firmware files (default=no).

```
acpi_override="{yes|no}"
```

[WARNING] ONLY USE THIS IF YOU KNOW WHAT YOU ARE DOING!

Override BIOS provided ACPI tables. For further documentation read

Documentation/acpi/initrd_table_override.txt in the kernel sources.

Search for ACPI table files (must have .aml suffix) in
acpi_table_dir= directory (see below) and add them to a separate
uncompressed cpio archive. This cpio archive gets glued
(concatenated, uncompressed one must be the first one) to the
compressed cpio archive. The first, uncompressed cpio archive is
for data which the kernel must be able to access very early (and
cannot make use of uncompress algorithms yet) like microcode or
ACPI tables (default=no).

acpi_table_dir="<dir>"

Directory to search for ACPI tables if acpi_override= is set to
yes.

early_microcode="{yes|no}"

Combine early microcode with ramdisk (default=yes).

stdloglvl="{0-6}"

Specify logging level for standard error (default=4).

Note

Logging levels:

- 0 - suppress any messages
- 1 - only fatal errors
- 2 - all errors
- 3 - warnings
- 4 - info
- 5 - debug info (here starts lots of output)
- 6 - trace info (and even more)

sysloglvl="{0-6}"

Specify logging level for syslog (default=0).

fileloglvl="{0-6}"

Specify logging level for logfile (default=4).

logfile="<file>"

Path to logfile.

sshkey="<file>"

SSH key file used with ssh-client module.

show_modules="{yes|no}"

Print the name of the included modules to standard output during build (default=no).

`i18n_vars="<variable mapping>"`

Distribution specific variable mapping. See `dracut/modules.d/10i18n/README` for a detailed description.

`i18n_default_font="<fontname>"`

The font `<fontname>` to install, if not specified otherwise. Default is "eurlatgr".

`i18n_install_all="{yes|no}"`

Install everything regardless of generic or host-only mode (default=no).

`reproducible="{yes|no}"`

Create reproducible images (default=no).

`noimageifnotneeded="{yes|no}"`

Do not create an image in host-only mode, if no kernel driver is needed and no `/etc/cmdline/*.conf` will be generated into the `initramfs` (default=no).

`loginstall="<directory>"`

Log all files installed from the host to `<directory>`.

`uefi="{yes|no}"`

Instead of creating an `initramfs` image, dracut will create an UEFI executable, which can be executed by an UEFI BIOS (default=no). The default output filename is

`<EFI>/EFI/Linux/linux-$kernel$-<MACHINE_ID>-<BUILD_ID>.efi`. `<EFI>` might be `/efi`, `/boot` or `/boot/efi` depending on where the ESP partition is mounted. The `<BUILD_ID>` is taken from `BUILD_ID` in `/usr/lib/os-release` or if it exists `/etc/os-release` and is left out, if `BUILD_ID` is non-existent or empty.

`machine_id="{yes|no}"`

Affects the default output filename of the UEFI executable, including the `<MACHINE_ID>` part (default=yes).

`uefi_stub="<file>"`

Specifies the UEFI stub loader, which will load the attached

kernel, initramfs and kernel command line and boots the kernel. The default is

```
/lib/systemd/boot/efi/linux<EFI-MACHINE-TYPE-NAME>.efi.stub.
```

```
uefi_splash_image="<file>"
```

Specifies the UEFI stub loader's splash image. Requires bitmap (.bmp) image format.

```
uefi_secureboot_cert="<file>", uefi_secureboot_key="<file>"
```

Specifies a certificate and corresponding key, which are used to sign the created UEFI executable. Requires both certificate and key need to be specified and sbsign to be installed.

```
kernel_image="<file>"
```

Specifies the kernel image, which to include in the UEFI executable. The default is `/lib/modules/<KERNEL-VERSION>/vmlinuz` or `/boot/vmlinuz-<KERNEL-VERSION>`.

```
enhanced_cpio="{yes|no}"
```

Attempt to use the dracut-cpio binary, which optimizes archive creation for copy-on-write filesystems (default=no). When specified, initramfs archives are also padded to ensure optimal data alignment for extent sharing. To retain reflink data deduplication benefits, this should be used alongside the `compress="cat"` and `do_strip="no"` parameters, with initramfs source files, tmpdir staging area and destination all on the same copy-on-write capable filesystem.

```
parallel="{yes|no}"
```

If set to yes, try to execute tasks in parallel (currently only supported for `--regenerate-all`).

FILES

```
/etc/dracut.conf
```

Old configuration file. You better use your own file in

```
/etc/dracut.conf.d/.
```

```
/etc/dracut.conf.d/
```

Any `/etc/dracut.conf.d/*.conf` file can override the values in

`/etc/dracut.conf`. The configuration files are read in

alphanumerical order.

AUTHOR

Harald Hoyer

SEE ALSO

dracut(8) dracut.cmdline(7)

dracut 057-4-gb46b3749

07/01/2022

DRACUT.CONF(5)